Unification of Temporary Storage in the NodeKernel Architecture

Patrick Stuedi, Animesh Trivedi, Jonas Pfefferle, Ana Klimovic, Adrian Schuepbach, Bernard Metzler
Managing Temporary Data

Input data

Broadcast
Map
Shuffle
Reduce

Intermediate data

Spark job

Tensorflow job

Output data

HDFS, S3

HDFS, S3

HDFS, S3
Running on Modern Hardware

Efficiently leveraging fast networking and storage hardware is difficult
Inflexible Software Stacks

Can’t implement every operation for all the different hardware and deployment options.
Flexible: Using an External Store

Integrate hardware once and support different operations and frameworks.
In the Talk...

• How should such a data store look like?
In the Talk...

• How should such a data store look like?
• Can we use existing distributed KV stores or filesystems?
In the Talk...

- How should such a data store look like?
- Can we use existing distributed KV stores or filesystems?
- NodeKernel: distributed storage architecture for temporary data storage designed for RDMA & NVMe Flash
  - Fast, flexible, easy to use from Spark, Flink, etc.